

# BUNDESREPUBLIK DEUTSCHLAND



## Prioritätsbescheinigung über die Einreichung einer Patentanmeldung

**Aktenzeichen:** 100 65 418.5

**Anmeldetag:** 27. Dezember 2000

**Anmelder/Inhaber:** Siemens Aktiengesellschaft, München/DE

**Bezeichnung:** Integrationsverfahren für Automatisierungs-  
komponenten

**IPC:** G 05 B 19/04

Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ursprünglichen Unterlagen dieser Patentanmeldung.

München, den 24. Juli 2001  
Deutsches Patent- und Markenamt  
Der Präsident  
Im Auftrag

Ebert

## Beschreibung

## Integrationsverfahren für Automatisierungskomponenten

5 Die Erfindung bezieht sich auf ein Verfahren zur Integration von mehreren Automatisierungskomponenten in ein einheitliches Ablaufebenenmodell eines jeweiligen Runtime-Systems einer industriellen Steuerung.

10 Es ist heutzutage üblich, sowohl für die speicherprogrammierbare Steuerung (SPS) als auch für die Bewegungssteuerung (MC) jeweils unterschiedliche hierarchische Ablaufebenen zu modellieren, denen Software-Tasks zur Steuerung des jeweiligen technischen Prozesses zugeordnet werden. Diese Tasks können  
15 Systemaufgaben erfüllen, sie können aber auch anwenderprogrammiert sein.

Aus DE 197 40 550 A1 ist es bekannt, dass Prozesssteuerungsfunktionalitäten der speicherprogrammierbaren Steuerungen

20 "SPS" und Bewegungsfunktionalitäten von MC-Steuerungen in einem einheitlichen konfigurierbaren Steuerungssystem integriert werden können.

Diese SPS/MC-Integration geschieht in Form des Zusammenschaltens von SPS- und MC-Steuerungsbaugruppen. Bei einer solchen  
25 Ausführung der Integration wird aber keine optimale und effiziente Taskstruktur für die Gesamtheit der Steuerungsaufgaben erreicht. Außerdem werden bei dieser Art der Integration hauptsächlich die klassischen MC-Funktionalitäten, wie sie  
30 insbesondere für Werkzeugmaschinen relevant sind, unterstützt. Anforderungen an die Steuerung, wie sie aus dem Betrieb von Produktionsmaschinen bekannt sind, werden durch diese Art des Zusammenschaltens von SPS- und MC-Steuerungsbaugruppen nicht optimal unterstützt.

35

In der Anmeldung DE 19 93 19 33.2 wird vorgeschlagen, den Takt des Kommunikationssystems zwischen dem PC-System und den

peripheren Geräten für einen Wechsel zwischen einem Echtzeitbetriebsprogramm und einem Nicht-Echtzeitbetriebsprogramm heranzunehmen. Hier ist es aber die Aufgabe dieser Taktabgreifung aus dem Kommunikationssystem, in einem industriellen Prozess einen möglichst reibungslosen Wechsel zwischen Echtzeit- und Nicht-Echtzeitanwendungen stattfinden zu lassen. Bei dieser Ausgestaltung wird der Grundtakt aber nur aus dem Takt des Kommunikationsmediums abgeleitet und er wird nur für den Wechsel des Betriebssystemmodus eines PC-Systems verwendet.

Der Erfindung liegt daher die Aufgabe zugrunde, für jeweils unterschiedliche Steuerungsaufgaben und unterschiedliche Randbedingungen bzw. Anforderungen des zugrunde liegenden technischen Prozesses in einfacher Weise optimale Ausprägungen einer industriellen Steuerung zu erstellen, die sowohl SPS- als auch MC-Funktionalität oder die Funktionalitäten weiterer Automatisierungskomponenten zur Verfügung stellt und somit auch für die Steuerung von Produktionsmaschinen geeignet ist.

Diese optimalen Ausprägungen werden prinzipiell durch ein einheitliches konfigurierbares Ablaufebenenmodell für die Steuerungs-Tasks der industriellen Steuerung erreicht.

Gemäß der Erfindung wird die oben genannte Aufgabe für ein Verfahren der eingangs genannten Art durch die folgenden aufeinander folgenden Schritte gelöst:

- a) Analyse der Ablaufeigenschaften der Automatisierungskomponenten,
- b) Ableiten einer Struktur für ein einheitliches Ablaufebenenmodell, bestehend aus einer Anordnung von System- und/oder Anwenderebenen, das die Anforderungen der beteiligten Automatisierungskomponenten abdeckt,

c) Zuordnung der System- und Anwendertasks in das Ablaufebenenmodell und

d) Programmieren der Anwenderebene bzw. der Anwenderebenen.

5

Ein Anwender kann dadurch auf eine definierte, systematische und flexible Art und Weise eine industrielle Steuerung mit integrativem Runtime-System erstellen, das Prozesssteuerungsfunktionalitäten von speicherprogrammierbaren Steuerungen  
10 "SPS", Bewegungsfunktionalitäten von "MC"-Steuerungen oder Funktionalitäten von weiteren Automatisierungskomponenten in sich vereint.

15

Durch die Schichtung in Systemebenen und Ablaufebenen wird der Kommunikationsaufwand zwischen den Tasks, die die Funktionalitäten der unterschiedlichen Automatisierungskomponenten repräsentieren, minimiert. Ein weiterer Vorteil liegt darin, dass die Programmierung der Steuerungsaufgaben für die unterschiedlichen Automatisierungskomponenten in einer einheitlichen Programmiersprache mit einer einheitlichen Erstelloberfläche erfolgen kann und dass der Anwender ein für seine jeweiligen Anforderungen zugeschnittenes Ablaufebenenmodell flexibel erstellen kann.

20

25

Ein weiterer Vorteil liegt darin, dass vom Anwender in die Anwenderebenen Anwenderprogramme ladbar sind. Dadurch kann der Anwender in seinen Anwenderprogrammen die Funktionalitäten der Automatisierungskomponenten sehr flexibel an die zugrundeliegenden Anforderungen des technischen Prozesses anpassen.  
30

35

Eine erste Ausgestaltung der vorliegenden Erfindung liegt darin, dass die System- und/oder Anwenderebenen priorisiert werden. Dadurch wird die Flexibilität des Anwenders bei der Erstellung eines integrativen Runtime-Systems einer industriellen Steuerung erhöht.

Eine weitere vorteilhafte Ausgestaltung der Erfindung liegt darin, dass der Grundtakt des Ablaufebenenmodells aus einem internen Timer (T2) oder aus einem internen Takt (T3) eines Kommunikationsmediums (B1, B2) oder aus einem externen Gerät (EG) oder von einer Größe (TG), die zum technologischen Prozess (P1, P2) gehört, abgeleitet wird. Dadurch kann sehr flexibel und sehr einfach der Grundtakt für das Ablaufebenenmodell abgeleitet werden. Dadurch, dass der Grundtakt für das Ablaufebenenmodell auch von einer Größe, die zum technologischen Prozess gehört, ableitbar ist, kann auf eine sehr einfache Weise eine direkte Rückkopplung aus dem technologischen Prozess zur Steuerung erhalten werden. Ein Anwender hat somit große Flexibilität, den Grundtakt zu erzeugen. Der Grundtakt kann z.B. aus äquidistanten Bussystemen wie z.B. Profibus sehr leicht abgeleitet werden.

Eine weitere vorteilhafte Ausgestaltung der Erfindung liegt darin, dass in die Anwenderebenen Anwenderprogramme ladbar sind. Dadurch kann der Anwender in seinen Anwenderprogrammen die Funktionalität der Steuerung sehr flexibel an die zugrunde liegenden Anforderungen des technischen Prozesses anpassen und er kann auch die Anwenderprogramme in unterschiedliche Anwenderebenen laden, um dadurch eine für seine jeweiligen Applikationen effektive Ausprägung der Steuerung zu erreichen.

Eine weitere vorteilhafte Ausgestaltung der Erfindung liegt darin, dass aus den Anwenderprogrammen ein programmierter Zugang zur Gesamtfunktionalität der Steuerung möglich ist. Der Anwender kann somit aus den Anwenderprogrammen heraus durch eine einheitliche Schnittstelle die Steuerungsfunktionalität nutzen.

Ein Ausführungsbeispiel der Erfindung ist in der Zeichnung dargestellt und wird im folgenden erläutert.

Dabei zeigen:

- FIG 1 in einer abstrakten Schemadarstellung Automatisierungskomponenten zum Betreiben technischer Prozesse,
- 5 FIG 2 die wesentlichen Ablaufebenen einer klassischen speicherprogrammierbaren Steuerung,
- FIG 3 die wesentlichen Ablaufebenen einer Bewegungssteuerung,
- 10 FIG 4 eine Schemadarstellung einer industriellen Steuerung,
- FIG 5 das integrative Ablaufebenenmodell des Runtime-Systems der industriellen Steuerung,
- 15 FIG 6 ein Ausführungsbeispiel für das Zuladen von Anwenderprogrammen in die Anwenderebenen und
- 20 FIG 7 in einer Schemadarstellung Möglichkeiten, wie der Grundtakt für die industrielle Steuerung gewonnen wird.

In der Darstellung gemäß FIG 1 sind Automatisierungskomponenten (SPS, MC, CNC, RC) zum Betrieb eines technischen Prozesses (P1, P2; FIG 4 bzw. FIG 7) dargestellt, die über eine Busverbindung BV miteinander verbunden sind. Durch jeweils 3 Punkte am linken und rechten Ende der Busverbindung BV ist angedeutet, dass weitere Automatisierungskomponenten an der Busverbindung BV hängen können. Darüber hinaus wird durch die Busverbindung BV auch der Anschluss an eine dazugehörige Steuerungseinheit (industrielle Steuerung) bzw. an Steuerungseinheiten hergestellt. Die Busverbindung BV ist nur exemplarisch gewählt, auch andere Kommunikationsmedien (z.B. Ringverbindungen oder wireless-Verbindungen sind denkbar). Die Automatisierungskomponenten (SPS, MC, CNC, RC) haben unterschiedliche Charakteristika und unterschiedliche Ablaufei-

genschaften. Die Analyse dieser Eigenschaften durch den Anwender kann über heuristische, empirische oder weitere auch kombinierbare Analysemethoden erfolgen. Reaktionszeiten und Zykluszeiten sind z.B. Parameter, die in diese Analyse mit-  
5 eingehen.

In der Darstellung gemäß FIG 2 sind die wesentlichen Ablauf-  
ebenen einer klassischen speicherprogrammierbaren Steuerung  
(SPS), angeordnet nach ihrer Priorität, gezeigt. Der Priori-  
10 tätsanstieg ist dabei durch einen Pfeil symbolisiert. In der  
niederpriorsten Ebene werden, wie durch eine gestrichelte Li-  
nie angedeutet, zwei unterschiedliche Aufgaben, nämlich ein  
freier Zyklus, d.h. "Anwenderebene freier Zyklus" und eine  
Hintergrund-Systemebene, d.h. "Systemebene Hintergrund" abge-  
15 wickelt. Der Hintergrund-Systemebene sind z.B. Kommunikati-  
onsaufgaben zugeordnet. Bei einer folgenden Anwenderebene,  
bezeichnet als "Anwenderebene zeitgesteuert", ist der Aufruf-  
takt der Tasks bzw. der Programme dieser Ebene parametrier-  
bar. Es erfolgt eine Überwachung dahingehend, ob die Bearbei-  
20 tung eines Anwenderprogrammes dieser getakteten Ebene recht-  
zeitig abgeschlossen ist, bevor das Starterereignis erneut auf-  
tritt. Läuft die Taktzeit ab, ohne dass das Anwenderprogramm  
der zugeordneten Ebene fertig abgearbeitet ist, wird eine  
entsprechende Task einer prioritätsmäßig übernächsten "Anwen-  
25 derebene für asynchrone Fehler" gestartet. In dieser "Anwen-  
derebene für asynchrone Fehler" kann der Anwender die Behand-  
lung von Fehlerzuständen ausprogrammieren.  
Auf die "Anwenderebene zeitgesteuert" folgt eine "Anwender-  
ebene Events". Die Reaktion auf externe oder interne Ereig-  
30 nisse (Events) erfolgt innerhalb der "Anwenderebene Events".  
Ein typisches Beispiel für ein solches Ereignis ist das  
Schalten eines binären bzw. digitalen Eingangs, wodurch typi-  
scherweise ein Ereignis ausgelöst wird. In einer "Systemebene  
hochprior" liegen die Aufgaben des Betriebssystems, welche  
35 die Arbeitsweise der programmierbaren Steuerung (SPS) sicher-  
stellen.

Die Darstellung gemäß FIG 3 zeigt die wesentlichen Ablaufebenen einer Bewegungssteuerung (MC). Auch hierbei sind die einzelnen Ebenen nach ihrer Priorität hierarchisch, wie durch einen Pfeil symbolisiert, angeordnet. Eine "Systemebene Hintergrund" und eine "Anwenderebene sequentiell" haben eine gleiche Priorität, nämlich die niedrigste. Diese aufgabenmäßige Zugehörigkeit ist wie bei FIG 2 durch eine gestrichelte Linie symbolisiert. Die Tasks der "Anwenderebene sequentiell" werden zusammen mit den Tasks der "Systemebene Hintergrund" im Round-Robin-Verfahren abgearbeitet. Typische Tasks der "Systemebene Hintergrund" sind z.B. solche für Kommunikationsaufgaben. In der "Anwenderebene sequentiell" laufen die vom Anwender programmierten Programmteile für die eigentliche Steuerungsaufgabe. Stößt die Steuerung in einem dieser Programmteile auf einen Bewegungs- oder Positionierbefehl, wird ein "Suspend" gesetzt, d.h. das Anwenderprogramm wird an dieser Stelle unterbrochen. In diesem Fall wird ein Befehl synchron genutzt. Die Abarbeitung dieses Bewegungs- oder Positionierbefehls geschieht in einer höchstpriorären "Systemebene getaktet". Ein jeder Lageregler oder Interpolator, der in der "Systemebene getaktet" abläuft, führt diesen Bewegungs- bzw. Positionierbefehl aus. Nach Ausführung des Befehls wird in die "Anwenderebene sequentiell" zurückgesprungen und das durch "Suspend" unterbrochene Anwenderprogramm wird durch ein "Resume" an der gleichen Stelle fortgesetzt. Die "Systemebene getaktet" enthält neben den schon erwähnten Lagereglern auch den Interpolationsteil der Steuerung.

Auf die niederpriorste Ebene setzt die "Anwenderebene Events" auf. Hier sind solche Tasks untergebracht, die auf externe oder interne Ereignisse reagieren. Solche Ereignisse können beispielsweise Alarmer sein.

In einer folgenden "Anwenderebene synchrongetaktet" laufen synchron getaktete Anwender-Tasks ab, z.B. Reglerfunktionalitäten. Diese Tasks sind synchronisiert zu getakteten System-



funktionen wie zum Beispiel Interpolator, Lageregler oder zyklische Buskommunikation.

5 In einer folgenden "Anwendererebene Events" sind solche Tasks untergebracht, die auf externe oder interne Ereignisse reagieren. Solche Ereignisse können beispielsweise Alarmer sein.

10 In der Darstellung gemäß FIG 4 wird in Form einer Schemadarstellung gezeigt, dass die Steuerung eines technischen Prozesses P1 über das Runtime-System RTS einer industriellen Steuerung erfolgt. Die Verbindung zwischen dem Runtime-System RTS der Steuerung und dem technischen Prozess P1 geschieht bidirektional über die Ein-/Ausgänge EA. Die Programmierung  
15 der Steuerung und damit das Festlegen des Verhaltens des Runtime-Systems RTS geschieht im Engineering-System ES. Das Engineering-System ES enthält Werkzeuge für die Konfigurierung, Projektierung und Programmierung für Maschinen bzw. für die Steuerung technischer Prozesse. Die im Engineering-System erstellten Programme werden über den Informationspfad I in das  
20 Runtime-System RTS der Steuerung übertragen. Bezüglich seiner Hardware-Ausstattung besteht ein Engineering-System ES üblicherweise aus einem Computersystem mit Graphikbildschirm (z.B. Display), Eingabehilfsmitteln (z.B. Tastatur und Maus), Prozessor, Arbeits- und Sekundärspeicher, einer Einrichtung  
25 für die Aufnahme computerlesbarer Medien (z.B. Disketten, CDs) sowie Anschlusseinheiten für einen Datenaustausch mit anderen Systemen (z.B. weiteren Computersystemen, Steuerungen für technische Prozesse) oder Medien (z.B. Internet). Eine  
30 Steuerung besteht üblicherweise aus Eingabe- und Ausgabeeinheiten, sowie aus Prozessor und Programmspeicher. Es ist auch vorstellbar, dass die Steuerung eines technischen Prozesses P1 über mehrere Runtime-Systeme RTS von industriellen Steuerungen erfolgt.

35

Die Darstellung gemäß FIG 5 zeigt das integrative Ablaufebenenmodell des Runtime-Systems (RTS; FIG 4) der industriellen

Steuerung. Die Priorisierung der Ebenen wird durch einen Pfeil in Richtung zur höchsten Priorität angedeutet. Die niederpriorsten Ebenen sind die "zyklische Anwenderebene" und die "sequentielle Anwenderebene". Diese beiden Ebenen laufen mit der gleichen Priorität. Deshalb sind diese Ebenen in der Darstellung gemäß FIG 4 durch eine gestrichelte Linie getrennt. Die "zyklische Anwenderebene" beinhaltet die "Background Task", die zykluszeitüberwacht ist. In der "sequentiellen Anwenderebene" werden die "Motion Tasks" durchlaufen. "Motion Tasks" sind nicht zykluszeitüberwacht und dienen im Wesentlichen zur Beschreibung sequentieller Abläufe. "Motion Tasks" werden quasiparallel abgearbeitet. Generell enthalten alle Anwenderebenen eine oder mehrere Tasks. Die Tasks nehmen die Anwenderprogramme auf. Die Tasks der "zyklische Anwenderebene" und der "sequentiellen Anwenderebene" werden in einem gemeinsamen Round-Robin-Zyklus abgearbeitet.

Die nächstfolgende Ebene ist die "zeitgesteuerte Anwenderebene". Die Tasks dieser Ebene werden zeitgesteuert aktiviert. Die Zeitsteuerung ist einer Granularität von Millisekunden einstellbar. Auf die "zeitgesteuerte Anwenderebene" folgt die "ereignisgesteuerte Anwenderebene". In dieser Ebene werden nach Erkennen eines User Interrupts so genannte "User Interrupt Tasks" aktiviert. User Interrupt Ereignisse können als logische Verknüpfung von Prozessereignissen und/oder internen Zuständen formuliert werden.

Die nächsthöhere Ebene ist die die "Anwenderebene für System Exceptions". In dieser "Anwenderebene für System Exceptions" werden System Interrupts überwacht, bei deren Eintreffen so genannte "Exceptions", d.h. Ausnahmefallbehandlungen, generiert werden. In der "Anwenderebene für System Exceptions" gibt es z.B. folgende Tasks, die bei Auftreten eines entsprechenden System Interrupts aktiviert werden:

- a) "Time Fault Task", die beim Ansprechen von Zeitüberwachungen aktiviert wird,

b) "Peripheral Fault Task", die z.B. bei Prozess- und Diagnosealarmen aktiviert wird, aber auch bei Stationsausfall oder Stationswiederkehr,

5 c) "System Fault Task", die bei allgemeinen Systemfehlern aktiviert wird,

d) "Program Fault Task", die bei Programmierfehlern (z.B. Division durch Null) aktiviert wird,

10

e) "Time Fault Background Task", die beim Ansprechen der Zykluszeitüberwachung der Background Task aktiviert wird und

f) "Technological Fault Task", die bei Technologiefehlern aktiviert wird.

15

Als nächstes folgt die Ebenengruppe "synchron getaktete Ebenen". Diese Ebenengruppe besitzt die höchste Priorität im Ablaufebenenmodell. Die einzelnen Ebenen dieser Ebenengruppe können untereinander weitere Priorisierungen aufweisen. Die Ebenengruppe "synchron getaktete Ebenen" besteht aus mindestens einer Systemebene und mindestens einer Anwenderebene. Die Systemebenen beinhalten die Systemfunktionen wie z.B. Lageregler oder Interpolator. In die Anwenderebenen dieser Ebenengruppe können von einem Anwender flexibel Anwenderprogramme (AP1 - AP4; FIG 6) zugeladen werden.

20

25

Für die Taktung der "synchron getakteten Ebenen" gibt es eine Reihe unterschiedlicher Taktgenerierungsmöglichkeiten. Der Grundtakt kann z.B. aus einem internen Timer (T1; FIG 7) kommen oder aus einem internen Takt (T3; FIG 7) eines Kommunikationsmediums (z.B. Profibus) oder aber der Takt kann auch aus einem Prozessereignis des technologischen Prozesses abgeleitet werden. Ein solches Prozessereignis kann z.B. die Taktrate (TG; FIG 7) eines Vorgangs an einer Produktionsmaschine oder Verpackungsmaschine sein. Anwenderebenen der Ebenengruppe "synchron getaktete Ebenen" können dabei basierend auf dem

30

35

Grundtakt getaktet werden, sie können aber auch synchron zu einer der Systemebenen der Ebenengruppe "synchron getaktete Ebenen" laufen. Die Anwendertasks dieser zu einer Systemebene synchronen Anwenderebene haben somit eine synchrone, d.h. deterministische Beziehung zu einer vom Anwender flexibel festlegbaren Systemebene. Das hat den Vorteil, dass deterministische Reaktionen auf Systemtasks (Systemtasks laufen in den Systemebenen), die der Anwender in seinen Anwendertasks programmiert hat, die in den Anwenderebenen der Ebenengruppe "synchron getaktete Ebenen" laufen, vom System garantiert werden. Das heißt z.B., dass das System garantiert, dass diese "synchrone Anwenderebene" entsprechend beispielhaft vor dem Interpolator aktiviert wird oder aber auch vor einer beliebigen anderen Systemfunktion.

Die "zeitgesteuerte Anwenderebene", die "ereignisgesteuerte Anwenderebene", die "sequentielle Anwenderebene", die "zyklische Anwenderebene" sowie die "Anwenderebene für System Exceptions" sind optional.

Die Task der "zyklischen Anwenderebene" (Background Task) ist zykluszeitüberwacht. Die "Motion Tasks" dagegen sind nicht zykluszeitüberwacht und dienen im wesentlichen zur Beschreibung sequentieller Abläufe. Das heißt, das vorliegende Ablaufebenenmodell unterstützt einen Anwender sowohl bei der Programmierung von sequentiellen Abläufen als auch bei der Ereignisprogrammierung. Es können somit synchrone Ereignisse als auch asynchrone Ereignisse durch die Programmierung erfasst werden. In die Anwenderebenen sind die vom Anwender erstellten Anwenderprogramme (AP1 - AP4; FIG 6) zuladbar. Die Anwenderprogramme AP1 bis AP4 werden üblicherweise mit Hilfe einer Programmierungsumgebung des Engineering-Systems (ES; FIG 4) erstellt.

Die Darstellung gemäß FIG 6 zeigt ein Ausführungsbeispiel für das Zuladen von Anwenderprogramm in die Anwenderebenen. FIG 6 zeigt exemplarisch eine Ausprägung von Anwenderebenen des Ab-

laufebenenmodells. Durch die drei Punkte am unteren Rand der Zeichnung ist dargestellt, dass auch noch weitere Anwender-  
ebenen, aber auch Systemebenen vorhanden sein können. Die  
Priorisierung der Ebenen wird wie im Vorangegangenen durch  
5 einen Pfeil in Richtung zur höchsten Priorität angedeutet.  
Den Anwenderebenen werden die Anwenderprogramme AP1 bis AP4,  
am rechten Bildrand durch kleine Rechtecke angedeutet, zuge-  
ordnet. Die Zuordnung wird dargestellt durch Zuordnungspfeile  
ZP1 bis ZP4. In den Anwenderebenen befinden sich Tasks, die  
10 die zugeladenen Anwenderprogramme AP1 bis AP4 aufnehmen. Die-  
se Tasks werden dann nach einer gewissen Strategie (z.B. se-  
quentiell) durchlaufen bzw. abgearbeitet.

Darstellung gemäß FIG 7 zeigt in einer Schemadarstellung Mög-  
lichkeiten, wie der Grundtakt für die industrielle Steuerung  
15 gewonnen wird. FIG 7 zeigt exemplarisch eine Kommunikations-  
topologie, in die die Steuerung S integriert ist. Die Steuer-  
ung S ist durch ein Rechteck dargestellt. Durch eine An-  
schlussleitung A2 ist die Steuerung S mit dem Bus B1 verbun-  
den, an dem über eine Anschlussleitung A1 das externe Gerät  
20 EG hängt. Über den Bus B2 erfolgt die Verbindung zum techni-  
schen Prozess P2. Der technische Prozess P2 ist am unteren  
Bildrand durch ein Rechteck dargestellt. Über die Anschluss-  
leitung A3 ist die Steuerung S mit dem Bus B2 verbunden, der  
wiederum über die Anschlussleitung A4 die Verbindung zum  
25 technischen Prozess P2 herstellt.

Die Generierung für den Grundtakt der Steuerung S kann aus  
unterschiedlichen Taktquellen erfolgen. So z.B. aus einer in-  
ternen Taktquelle, dargestellt durch den internen Timer T2  
30 der Steuerung S oder auch durch eine externe Taktquelle wie  
z.B. den Timer T1, der zum externen Gerät EG gehört. Als ex-  
terne Taktquelle kann aber auch der Grundtakt eines Kommuni-  
kationsmediums dienen. Wenn der Bus B2 z.B. durch einen äqui-  
distanten Profibus realisiert wird, dann kann der Takt für  
35 die Steuerung aus dem Grundtakt dieses Busses gewonnen wer-  
den. In FIG 7 ist dies dargestellt dadurch, dass der Timer T3  
direkt an der Anschlussleitung A3 positioniert ist, und diese

Anschlussleitung A3 stellt die Verbindung zum Bus B2 her. Die Steuerung hängt somit als Slave am Bus und kann direkt den Bustakt verwenden. Es gibt mehrere Varianten, wie der Takt für die Steuerung aus dem Grundtakt eines Kommunikationsmediums (z.B. eines Busses) gewonnen werden kann:

5 Zum einen kann die Steuerung S Slave am Bus sein, die Taktinformation kommt dann von extern über den Bus. Zum anderen kann die Steuerung S Master am Bus sein. Die Taktquelle liegt in diesem Fall in der Steuerung S. Für diesen Fall existieren  
10 zwei Ausprägungen. Die Taktquelle kann in einer Masterbusanschaltung liegen oder die Taktquelle ist in der Steuerung S, hierbei erfolgt die Einspeisung des Taktes in die Masterbusanschaltung.

15 Weiterhin kann als externe Taktquelle ein Taktgeber TG dienen, der im technischen Prozess P2 integriert ist. Ein Taktgeber TG in einem technischen Prozess kann z.B. der Arbeitstakt einer Produktionsmaschine oder Verpackungsmaschine sein. In der Darstellung gemäß FIG 7 sind als Kommunikationsmedien  
20 beispielhaft Busverbindungen dargestellt. Es können aber als Kommunikationsmedien aber auch Ring-, Stern- oder andere Verbindungsarten gewählt werden, auch wireless-Verbindungen. Aus diesen Verbindungssystemen kann dann der oben genannte Grundtakt abgeleitet werden.

## Patentansprüche

1. Verfahren zur Integration von mehreren Automatisierungskomponenten in ein einheitliches Ablaufebenenmodell eines jeweiligen Runtime-Systems (RTS) einer industriellen Steuerung (S),

g e k e n n z e i c h n e t d u r c h folgende aufeinander folgenden Schritte:

- 10 a) Analyse der Ablaufeigenschaften der Automatisierungskomponenten (SPS, MC, RC, CNC),
- b) Ableiten einer Struktur für ein einheitliches Ablaufebenenmodell, bestehend aus einer Anordnung von System- und/oder Anwenderebenen, das die Anforderungen der beteiligten Automatisierungskomponenten (SPS, MC, RC, CNC) ab-
- 15 deckt,
- c) Zuordnung der System- und Anwendertasks (AP1 - AP4) in das Ablaufebenenmodell und
- d) Programmieren der Anwenderebene bzw. der Anwenderebenen.

20 2. Verfahren zur Integration von mehreren Automatisierungskomponenten nach Anspruch 1,  
d a d u r c h g e k e n n z e i c h n e t,  
dass die System- und/oder Anwenderebenen priorisiert werden.

25 3. Verfahren zur Integration von mehreren Automatisierungskomponenten nach Anspruch 1 oder Anspruch 2,  
d a d u r c h g e k e n n z e i c h n e t,  
dass der Grundtakt des Ablaufebenenmodells aus einem internen Timer (T2) oder aus einem internen Takt (T3) eines Kommunikationsmediums (B1, B2) oder aus einem externen Gerät (EG) oder

30 von einer Größe (TG), die zum technologischen Prozess (P1, P2) gehört abgeleitet wird.

35 4. Verfahren zur Integration von mehreren Automatisierungskomponenten nach einem der vorstehenden Ansprüche  
d a d u r c h g e k e n n z e i c h n e t,

dass in die Anwendererebenen Anwenderprogramme (AP1 - AP4) ladbar sind.

5. Verfahren zur Integration von mehreren Automatisierungskomponenten nach einem der vorstehenden Ansprüche
- 5     d a d u r c h     g e k e n n z e i c h n e t ,  
dass aus den Anwenderprogrammen (AP1 - AP4) ein programmierter Zugang zur Gesamtfunktionalität der Steuerung möglich ist.



## Zusammenfassung

## Integrationsverfahren für Automatisierungskomponenten

- 5 Verfahren zur Integration unterschiedlicher Automatisierungs-  
komponenten (SPS, MC, RC, CNC) in eine industrielle Steuerung  
(S) mit einem einheitlichen Ablaufebenenmodell des Runtime-  
Systems (RTS). Der Grundtakt des Ablaufebenenmodells wird da-  
bei aus einem internen Timer (T2) oder aus einem internen  
10 Takt (T3) eines Kommunikationsmediums (B1, B2) oder aus einem  
externen Gerät (EG) oder von einer Größe (TG), die zum tech-  
nologischen Prozess (P1, P2) gehört, abgeleitet.

FIG 1

1/4

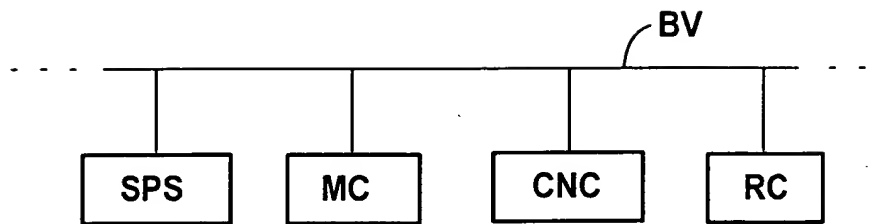


FIG 1

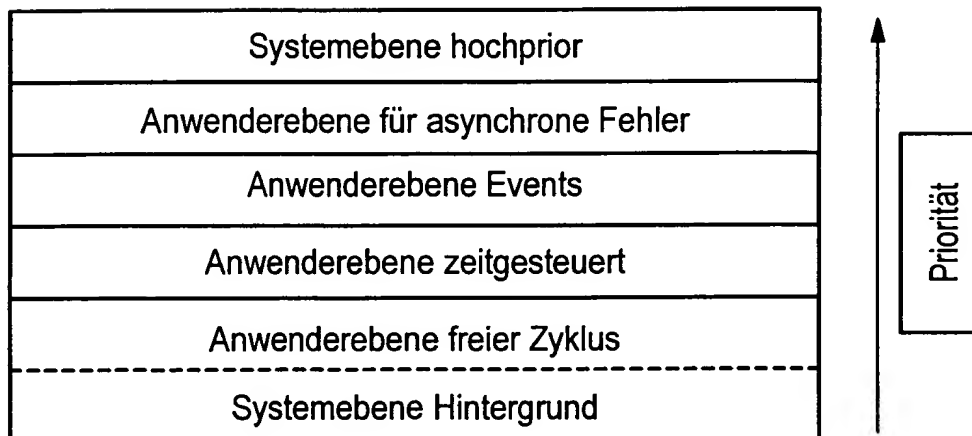


FIG 2

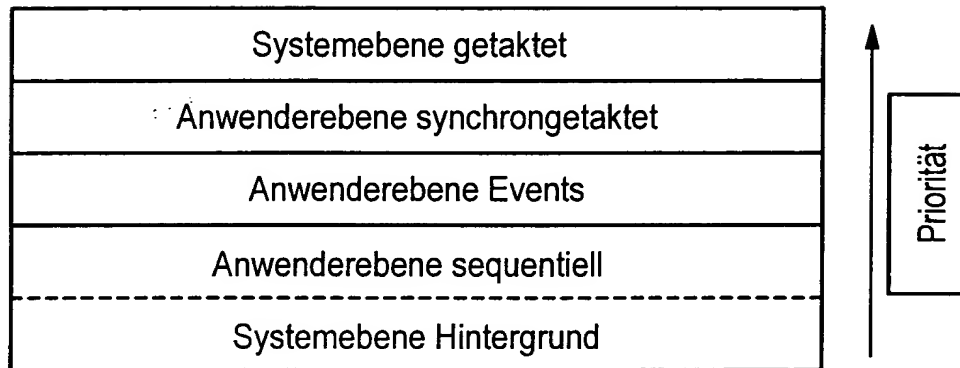


FIG 3

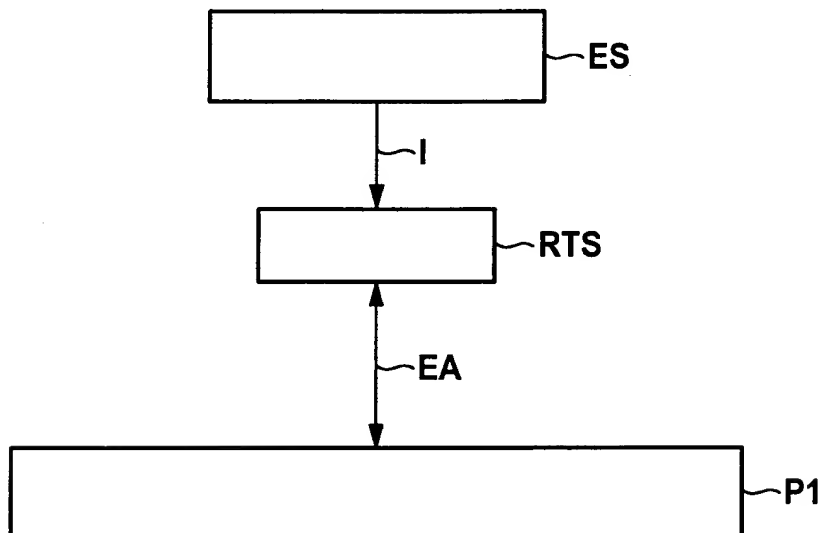


FIG 4

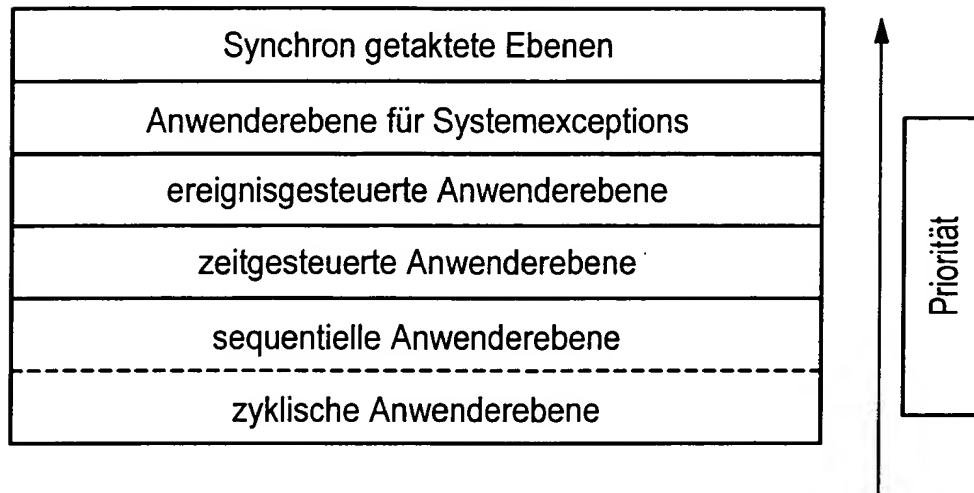


FIG 5

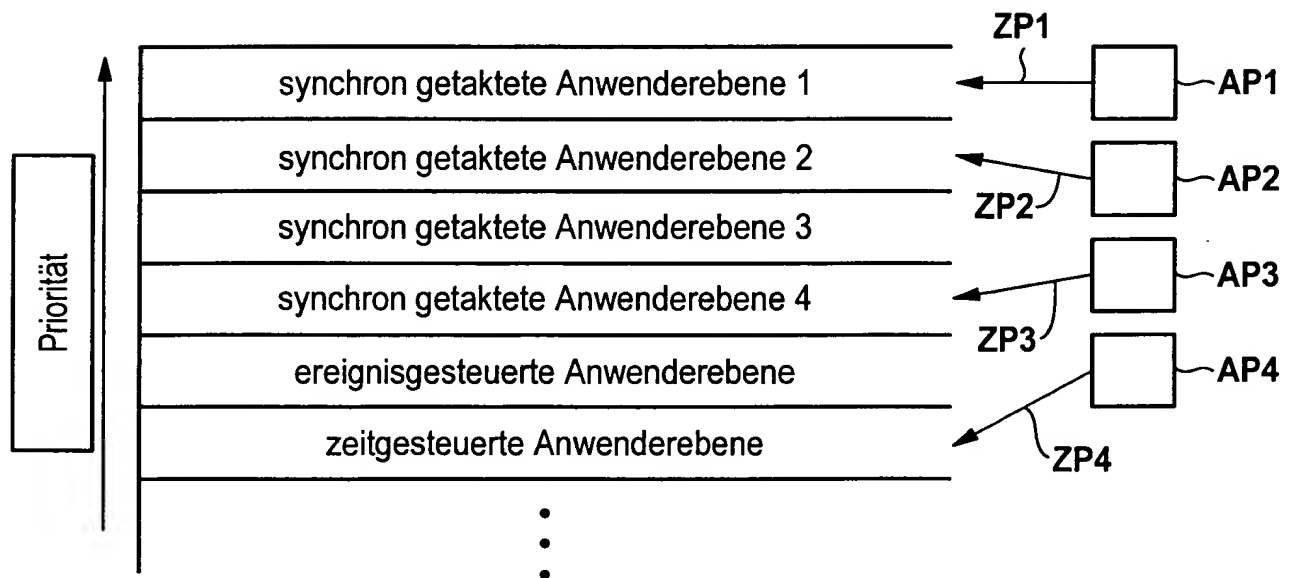


FIG 6

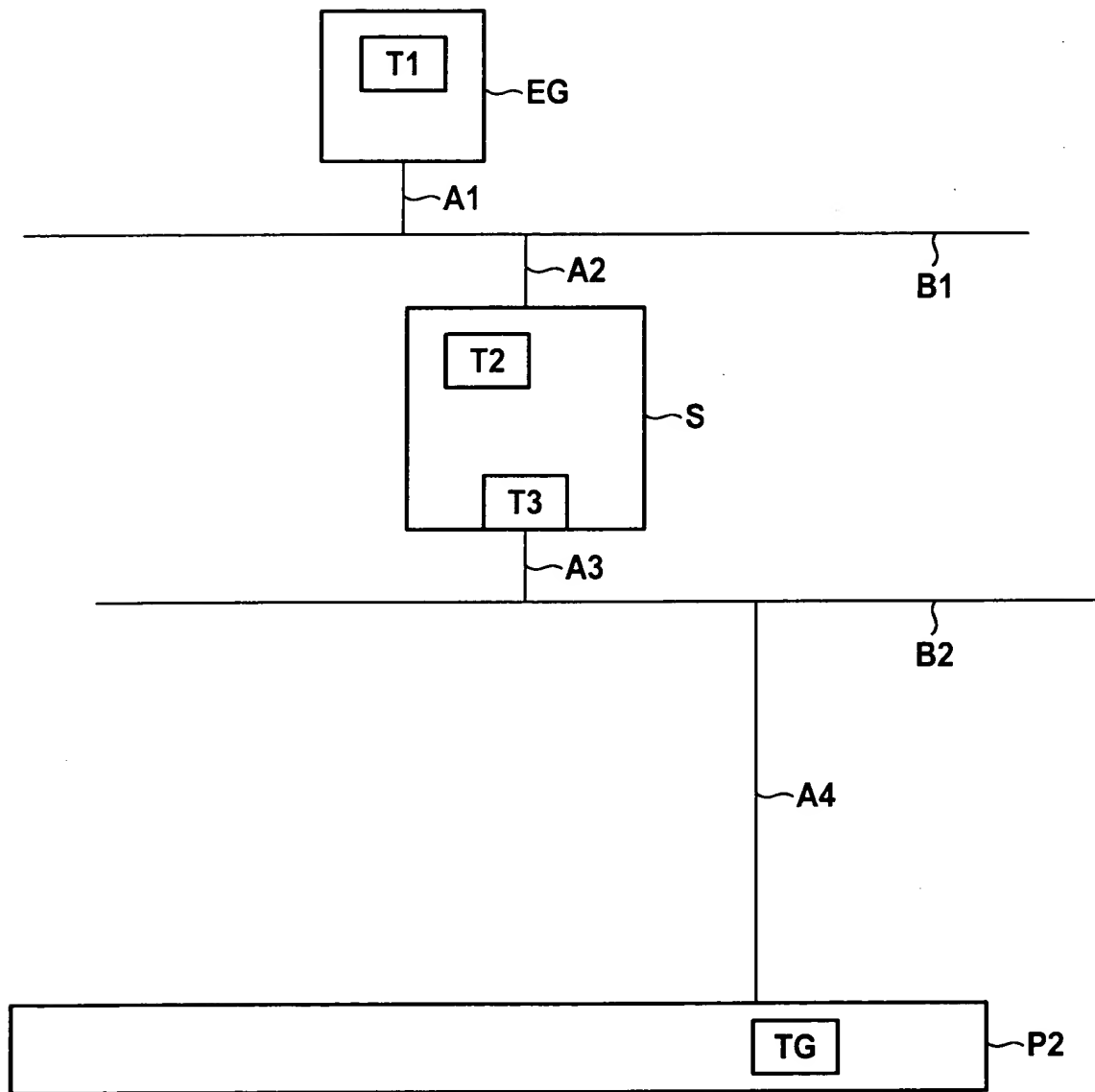


FIG 7